

MODIS SCIENCE DATA SUPPORT TEAM PRESENTATION

February 21, 1992

AGENDA

1. Action Items
2. SDST Total Quality Management (TQM)
3. MODIS Airborne Simulator (MAS)
4. Cloud Algorithm Porting
5. SDST Schedule

ACTION ITEMS:

08/30/91 [Lloyd Carpenter and Team]: Draft a schedule of work for the next 12 months. Include primary events and milestones, documents to be produced, software development, MAS support, etc. (An updated draft schedule is included in the handout.) STATUS: Open. Due date 09/27/91.

12/06/91 [Liam Gumley]: Investigate a cataloguing scheme for the MAS data. Consider the Master Catalogue, PLDS and PCDS. (A proposed scheme is included in the handout.) STATUS: Open. Due date 02/14/92.

12/06/91 [Liam Gumley, Tom Goff, Ed Masuoka]: Develop a plan for storing and distributing MAS data. (A proposed plan is included in the handout.) STATUS: Open. Due date 02/14/92.

01/03/92 [Ed Masuoka]: Check on the UCAR "copyright" as a first step in standardizing an SDST software copyright statement for code sharing. Check with legal. (The proposed notice is included in the handout.) STATUS: Open. Due date 02/14/92.

01/03/92 [Team]: Check on the set of software engineering tools available in Code 530 to see if any of these would be of use to the SDST. (We left a message with Julie Breed, Code 563.2, to see if we can arrange to run the Cloud Algorithm through their Pro:QA.) STATUS: Open. Due date 02/14/92.

01/17/92 [Tom Goff]: Have a polished version (with peer review) of the file dump routine ready for the MODIS Science Team Meeting. STATUS: Open. Due date 04/01/92.

NOTICE

FDUMP.C, Version 1.1, February, 1992

Sponsor: National Aeronautics and Space Administration (NASA)
Goddard Space Flight Center (GSFC)
Moderate Resolution Imaging Spectrometer (MODIS)
Science Data Support Team (SDST)
Code 920.0, Greenbelt, MD 20771

Author: Research and Data Systems Corporation (Thomas E. Goff)
7855 Walker Drive, Suite 460, Greenbelt, MD 20770
(301) 982-3704, Tgoff on GSFC Mail
teg@ltpiris2.gsfc.nasa.gov

This software may be freely used and distributed without any compensation to the author or the sponsor. It is provided without support and without any obligation, whatsoever, to assist in its use, correction, modification or enhancement.

THIS SOFTWARE IS PROVIDED "AS IS" WITH NO EXPRESS OR IMPLIED WARRANTIES OF ANY KIND, INCLUDING WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. There is no warranty that this software will meet any particular specification nor is there any warranty that the documentation providing instructions or information for use of the software is accurate or otherwise conforms to the software requirements. Further, in furnishing this software, there shall be no liability, under any circumstances, for either direct or consequential damages.

Any user of this software agrees that they will repeat this Notice, in its entirety, prior to the distribution of this software to another.

*(This version of the MODIS SDST Software Notice was provided
by Ron Sandler of the GSFC Patent Counsel's Office.)*

Standardized Copyright Notice
Thomas E. Goff
20 February, 1992

tgoff on GSFC mail,
teg@ltpiris2.gsfc.nasa.gov,
or (301) 982-3704

We are attempting to standardize our copyright (or other) notice to be included with all software that is placed into the public domain, either via anonymous ftp or BBS's. Here is the latest version that consists of a few additional words added to the legal notice provided by Ran Sandler of the GSFC legal office. Our goal is to provide useful software while not incurring any legal problems.

MODIS SDST Legal Notice

```
/*Legal ***** NOTICE ***** NOTICE ***** NOTICE*****
*
*   This software may be freely used and distributed without any compensation
* to the author or the sponsor. It is provided without support and without any
* obligation, whatsoever, to assist in its use, correction, modification, or
* enhancement.
*
*   THIS SOFTWARE IS PROVIDED "AS IS" WITH NO EXPRESS OR IMPLIED WARRANTIES
* OF ANY KIND, INCLUDING WARRANTIES OF DESIGN, MERCHANTABILITY, OR FITNESS FOR
* A PARTICULAR PURPOSE; OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE
* PRACTICE. There is no warranty that this software will meet any particular
* specification nor is there any warranty that the documentation providing
* instructions or information for use of the software is accurate or otherwise
* conforms to the software requirements. Further, in furnishing this software,
* there shall be no liability, under any circumstances, for either direct or
* consequential damages.
*
*   Users of this software agree that they will retain this Notice, in its
* entirety, within the software source listing and documentation, prior to the
* distribution of this software to other parties.
*
*****c*/
```

MODIS SDST Authorship Notice

```
/*Author, sponsor: *****
*
*   National Aeronautical and Space Administration (NASA)
*   Goddard Space Flight Center (GSFC)
*   Moderate Resolution Imaging Spectrometer (MODIS)
*   Science Data Support Team (SDST)
*   code 920.2, Greenbelt, MD, U.S.A. 20771
*
*   Research and Data Systems Corporation (RDC)
*   7855 Walker Drive, Suite 460, Greenbelt, MD, 20770-7777
*   Thomas E. Goff, (301) 982-3704, tgoff on GSFC mail,
*   or teg@ltpiris2.gsfc.nasa.gov on the internet
*
*****a*/
```

Note: The asterisk (*) in column 1 can be used in most FORTRAN compilers to designate a comment line. The /* however may be interpreted as a divide and multiply if improperly placed!

MODIS SDST APPLICATIONS OF TOTAL QUALITY MANAGEMENT (TQM)

- Defining TQM in a Software Environment
- TQM's Underlying Principles
- TQM's Quality Pyramid
- Applying TQM to the MODIS SDST Software Lifecycle
- Current MODIS SDST Activities Already Including TQM Techniques
- What Changes/Accommodations Might be Necessary for Further Applying TQM in MODIS SDST Software Activities?

DEFINING TQM IN A SOFTWARE ENVIRONMENT

- A team-wide process with its own products that affects the actual processes and products of the MODIS SDST software lifecycle.
- Quality is realized by building it into the application system to prevent undesirable results, rather than trying to inspect, identify, and modify problem software after the fact.
- An awareness resulting from management and technical training and application.
- Emphasizes eliminating bug sources, not just bug detection.
- Increased up-front costs predicated on eliminating even higher future costs, such as bug correction and on-going maintenance.
- Inclusive lifecycle methodologies involving ECS, MODIS, and MODIS SDST components, their inputs, interfaces, and outputs.

TQM's QUALITY PYRAMID

- TQM at the base means ECS/MODIS SDST management recognizes its responsibility for application system and software quality. At least 90 percent of all quality problems are attributable directly to, and can only be corrected by, management.
- Quality Control in the middle, measures the developed software to determine whether or not it satisfies standards and requirements. This is implemented through design and code walkthroughs, reviews and inspections, multi-level build testing, and multiple levels of acceptance testing. Manual and automated tools are used.
- Quality Assurance at the top is the continual process of L1A/B quality improvement. QA collects, summarizes, and analyzes any defects to find their root causes in the overall design and development process. Eliminating the causes eliminates defects, and quality improves.



Adapted from: William E. Perry, "Improving Software Development," Signal, January, 1990, pages 59-63.

TQM's UNDERLYING PRINCIPLES--THE FOURTEEN COMMANDMENTS

(Note: These date from the early 1950s.)

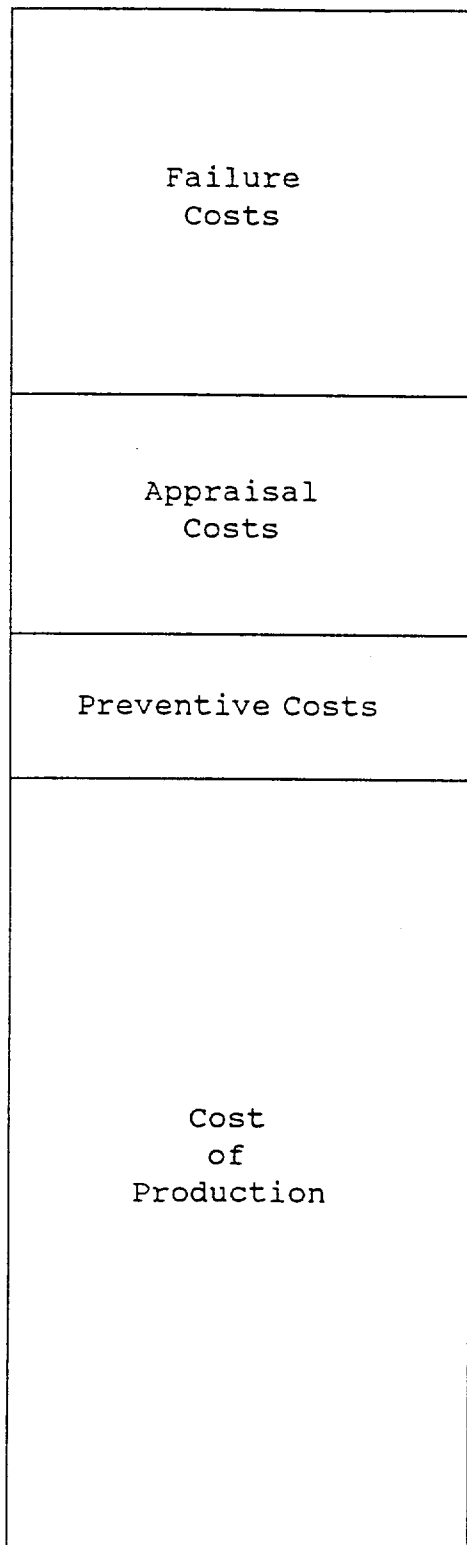
- Create constancy of purpose for improvement of product and service.
- Adopt the new philosophy. We are in a new economic age. Management must awaken to the challenge, must learn their responsibilities, and take on leadership for change.
- Cease dependence on inspection to achieve quality. Instead, build quality into products and services in the first place.
- End the practice of awarding business on the basis of price tag alone.
- Improve constantly and forever every process for planning, production, and service.
- Institute training on the job.
- Adopt and institute leadership.
- Drive out fear, so that everyone may work effectively for the company.
- Break down barriers between staff areas.
- Eliminate slogans, exhortations, and targets for the work force.
- Eliminate numerical quotas for the work force and numerical goals for management. Substitute leadership.
- Remove barriers that rob people of pride of workmanship. Eliminate the annual rating or merit system.
- Institute a vigorous program of education and self-improvement for everyone.
- Put everyone in the company to work to accomplish the transformation.

Adapted from: W. Edwards Deming, Out of Crisis, MIT Center for Advanced Engineering Study, 1986.

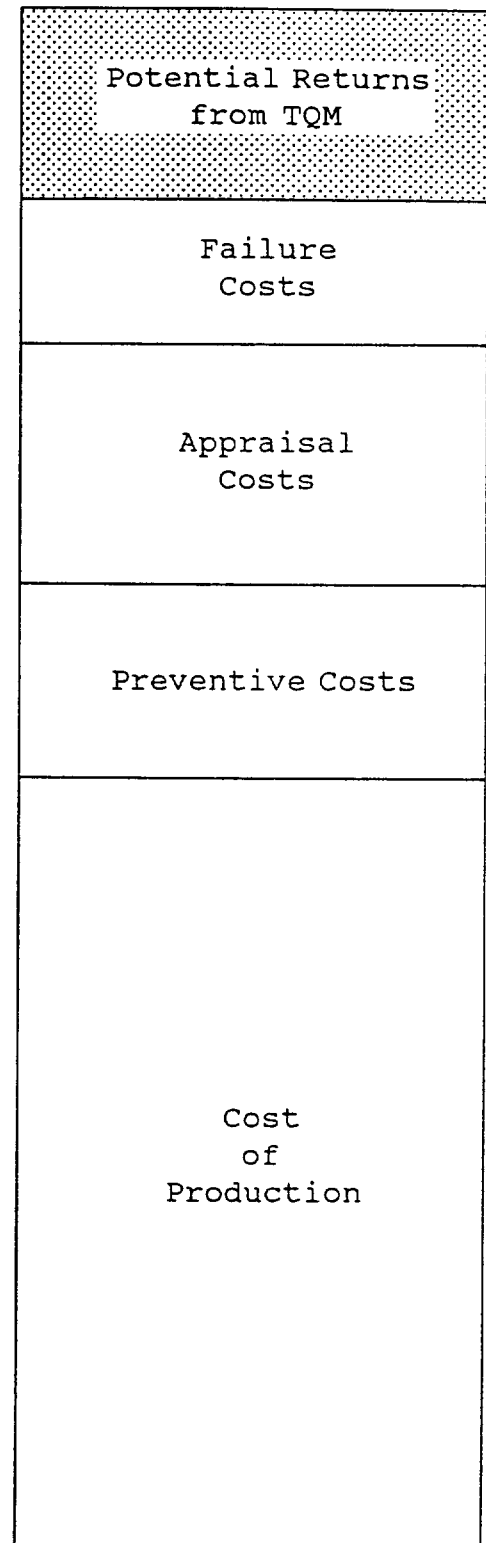
APPLYING TQM TO THE MODIS SDST SOFTWARE LIFECYCLE

- At ECS' inception:
 - Documenting the performance, functional, and interface requirements for each ECS software component, and
 - Specifying the PGS software and hardware environments.
- Walkthroughs of successive levels of ECS documentation to surface defects and deficiencies so they can be corrected as early as possible.
- Prototyping on a specified testbed to clarify requirements, prove feasibility, and obtain user feedback.
- Providing a software design and development environment with COTS and/or developed tools to minimize the introduction of error.
- Sustained use of performance metrics and development standards in a controlled production environment.
- Documentation, development, and testing standards for PGS production software originating from other sources.

Cost to Build Software



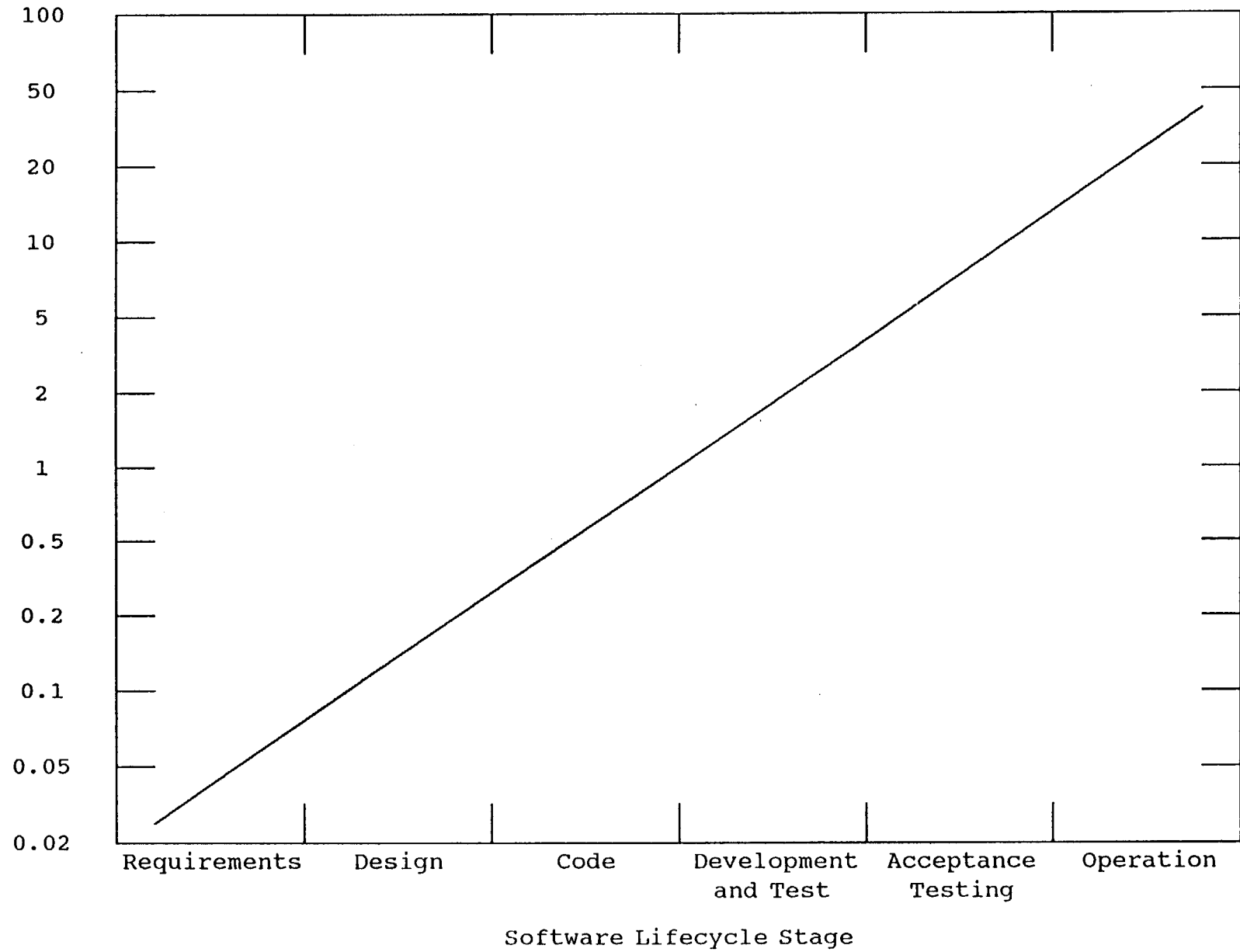
Cost to Build Software After Implementing TQM



The application of TQM for MODIS SDST software development results in higher preventive costs up-front. They are more than offset by a reduction of failure costs. The net effect is lower lifecycle cost.

Relative Cost of Software Errors

Relative Cost
to Fix Error



CURRENT MODIS SDST ACTIVITIES ALREADY INCLUDING TQM TECHNIQUES

- L1A/B functional and performance requirements are traceable to ECS and/or MODIS requirements.
- Front-end software design activities include design reviews and prototyping.
- Using MODIS SDST-wide standardized methodologies such as CASE tools and a specified HOL (to the extent their final selection can be anticipated).
- Recommending a set of standardized methodologies for non-MODIS SDST software development and acceptance, including impacts from architecture differences among development and production sites.
- Prototyping with test data having known results--precursor datasets.
- Reusing proven algorithms.

ADDITIONAL TQM TECHNIQUES FOR MODIS SDST SOFTWARE ACTIVITIES

- Preparing a MODIS SDST software quality statement to include:
 - Quality-oriented responsibilities at each ECS, MODIS, and SDST management and technical level,
 - The L1A/B development environment with its toolkits and procedures, and
 - The need for acceptance criteria and procedures for MODIS SDST-developed and other-developed software.
- Providing a quality-related orientation to persons involved in software design, development, and implementation, emphasizing their contributions to the implementation of quality L1A/B software.
- Specifying how the three Quality Pyramid components will be implemented on an on-going basis.
- MODIS SDST participation in activities such as reviews of functional or performance specifications from which MODIS SDST software design, development, and implementation requirements are derived.
- Performing a full set of L1A/B software functional and performance requirements reviews, including the SDR with its feasibility demonstrations and resulting allocated (build-to) baseline(s).
- Specifying the QA/QC/CM tools and procedures (such as module-level and build testing) required for designing, developing, and implementing quality software.
- Providing and maintaining a controlled development environment that includes:
 - COTS or developed CASE tools,
 - Standardized I/O and other common routines,
 - Mathematical libraries,
 - Performance metrics, and
 - Control of the Builds of developed software.
- Further implementing acceptance procedures for other software, such as scientist-developed algorithms, that either provides input to MODIS SDST software or uses MODIS SDST-generated output in the PGS.

Sources Consulted

NASA Total Quality Management 1989 Accomplishments Report, NASA Safety and Mission Quality Office, Washington, D.C., 1990.

Benchmarking, Sharon Sloane Enterprises, Inc., Rockville, MD.

MODIS Science Data Support Team (SDST) Schedule, February 12, 1992.

W. Edwards Deming, Out of Crisis, MIT Center for Advanced Engineering Study, 1986.

Robert H. Lochner, "A Sequential Process for Total Quality Management," Proceedings of the IEEE 1989 National Aerospace and Electronics Conference NAECON 1989, Volume 4, pages 1641-1644.

Thomas J. Murrin, "Quality is the Key to Technical Excellence and Competitiveness," Signal, January, 1990, pages 29-34.

Joyce R. Jarrett, "NASA's Approach to Quality and Productivity Improvement," Signal, January, 1990, pages 65-68.

William E. Perry, "Improving Software Development," Signal, January, 1990, pages 59-63.

Investigation of a cataloguing scheme for the MODIS Airborne Simulator (MAS)

Liam Gumley, RDC

20 February 1992

The MODIS Airborne Simulator (MAS) is currently providing image data for the MODIS Science Team and will continue to be a major source of data up to and beyond the launch of the first MODIS-N instrument in 1998. As with any remote sensing data set, one issue that must be addressed for the MAS data is that of user accessibility. Users must be able to identify, locate, and retrieve relevant data without encountering major obstacles. A data catalogue is a major requirement for this process to occur.

Catalogue System Definition and Requirements

The Committee on Earth Observations Satellites (CEOS) Working Group on Data has published a set of definitions and requirements for a data catalogue ("Guidelines for an Internationally Interoperable Catalogue System", Issue 1.0, November 1991, Section 1.4). Contributors to this document included (among others) Mary James, Ken McDonald, Lola Oleson, and Jim Thieman (GSFC).

'A catalogue system provides a service. It enables a user to obtain detailed information about whole data sets; typically specific to a discipline, data center, or project. A catalogue also allows a user to identify and retrieve individual granule(s) (the smallest part of the data set retrievable from the archive) of the data set, by specifying independent variable range(s). Having identified a set of granules which may be of value a user should be able to review the contents of a granule (browse, quicklook) and place an order for one or more granules.

A catalogue is assumed to have three main components:

- Directory Service: Provides descriptions of metadata or data set catalogues containing high level information suitable for making an initial determination of the potential usefulness of a data set for some application. Information on the location of metadata or data set catalogues will be found in this directory.
- Guide Service: Provides detailed information concerning specific data sets which enable the user to make a detailed analysis of whether a data set or a specific granule within the data set will be of value for some application. May also contain information necessary for analysis of the data (e.g. calibration coefficients).
- Inventory Service: The inventory service contains information needed to identify and retrieve the individual granule(s) of the data set, given the specification of the independent variable range(s); may contain information extracted from the data set granules (e.g. % cloud cover) as well as information to enable ordering (e.g. price).'

Catalogue Users

The CEOS Working Group also defined a number of different categories of data users, not all of which may be applicable to MAS data ("Guidelines for an Internationally Interoperable Catalogue System", Issue 1.0, November 1991, Section 2.1). It is envisioned that MAS data users will have varying requirements in terms of data types, availability, timeliness of retrieval and so on. Based on the classifications of data users developed by CEOS, the main groups of users associated with the MAS are classified as follows:

MODIS Science Team Members

These users are developing the algorithms for the MODIS-N instrument, and therefore will be the prime users of MAS data. They will decide the mission schedules for the MAS in terms of location, time of year, coverage, spectral bands used etc. It is clear that this group of users requires the greatest level of support in the development of a MAS data catalogue. A feature of this group is their diversity in specialization (e.g. Atmosphere, Land, Ocean) and their geographic separation (some are on site at GSFC, most are spread over locations throughout the USA). An example would be a MODIS Atmosphere Group member requiring cloudy MAS flight tracks over an uplooking IR interferometer deployed on the ground during a field campaign.

Other EOS Science Team Members

Members of other EOS Science Teams (e.g. ASTER) may require access to MAS data for their own algorithm development efforts. While they are not expected to plan missions for the MAS, it is conceivable that they will be interested in specific missions and data sets. An example would be an ASTER team member requiring cloud-free MAS data collocated with a Landsat overflight.

General University/Institutional Users

As the MAS instrument matures, it is likely that users outside the EOS program will require access to MAS data sets in support of their own research programs. An example would be a co-investigator in a MAS field campaign who runs a ground based instrument (e.g. lidar) who desires imagery of clouds from MAS.

While the MAS data may eventually attract broader interest, it is envisioned that in the near term (1-3 years) most users of MAS data will fall into these groups.

Catalogue User Services

The following user services are suggested examples of what would be required from a catalogue system. These requirements assume an interactive, remotely accessible catalogue system.

- Identification of MAS data suitable for a particular application based on
 - geographic location/coverage,
 - spectral band availability,
 - cloud cover,
 - land/water coverage,
 - solar elevation,
 - collocated satellite passes,
 - collocated in-situ data (ground or aircraft),
 - calibration availability/quality,
 - data noise estimates.
- Retrieval and/or visualization of browse/quick-look products
 - for real time viewing,
 - or for downloading to users own local site (for off-line viewing).
- Placement of requests for data
 - for on-line retrieval (by FTP for example),
 - or for offline retrieval (obtaining a magnetic tape copy).
- Notification of current instrument status regarding
 - instrument modifications,
 - scheduled missions,
 - spectral band characteristics (e.g. spectral responses).

With reference to the three main components of a catalogue listed earlier, it seems that most of these functions fit into the category of Guide and Inventory Services. The most fundamental property of the catalogue should be the ability to identify the availability of specified datasets for a given application (Inventory Service). The Guide Service would comprise a small part of the catalogue, although it would require regular updates to keep users aware of changes in the MAS instrument or program.

Guide/Inventory Description

The items to be included in an inventory depend on the type of sensor being described. For an airborne sensor such as the MAS, this introduces the concept of flight lines (tracks). A flight line (track/segment) is when the aircraft flies straight and level at cruising altitude for some period of time, acquiring at least several hundred scan lines of image data. Individual flight lines are the "granules" (i.e. smallest independently identified items) for the purposes of MAS processing, cataloguing, and distribution.

The MAS catalogue should have two major components. The first would be the MAS Guide Service. The second would be the MAS Inventory Service. Both of these could be hosted by a relational DBMS, with a user interface to provide the required query paths into the various tables. The following description of this kind of catalogue is based on work done by Carroll Hood (RDC) in specifying the design of a catalogue for two airborne scanners similar to the MAS, the TIMS (Thermal Infrared Multispectral Scanner) and the CAMS (Calibrated Airborne Multispectral Scanner). Both of these are operated by Stennis Space Center.

The MAS Guide/Inventory would be structured in a relational DBMS as follows.

A Guide layer application would contain 2 free-form text files. These would be

■ *DESCRIPTION OF THE INSTRUMENT*

- Full instrument name
- Instrument objectives
- Principal Investigator (name, address, phone, email)
- Home base (when not on deployment)
- Aircraft platform name
- Nominal aircraft altitude
- Scan angular width
- Instantaneous field of view (angular and nominal spatial)
- Scan rate
- Number of pixels per scan
- Digitization levels per pixel
- Data rate
- Calibration sources
- Modification history
- Mission planning contact person (name, address, phone, email)
- Level-1 data processing contact person (name, address, phone, email)

■ *MISSION SCHEDULE*

- Instrument missions prior to current date (mission date, name, purpose)
- Next mission following current date
- Planned future missions

The Inventory application would use 5 tables in a single relational database. These would be

- **MISSION**

A Mission is a sequence of one or more flights based at a certain location for specific purpose.

- **FLIGHT**

A single flight has one or more flight lines (tracks). It should be noted that data is still available at Level-0 between flight lines (i.e. during aircraft turns and ascent/descent) however it is not routinely processed.

- **LINE**

A single flight line is defined as the inventory granule for a given flight.

- **WHO**

Information on mission PIs, and support personnel who may have performed instrument setup or calibration in the field, or Level-1 processing after a mission.

- **REFERENCE**

Information on publication references resulting from analysis of instrument data sets.

Inventory Table Structures

- **MISSION Table**

Mission Number
Mission title (acronym expanded)
Duration (start and end dates)
Purpose
Location (country, state)
Principal Investigator
Number of Flights

- **FLIGHT Table**

Mission Number
Flight Number
Date
Data acquisition location
Takeoff location
Takeoff time
Landing time
Aircraft Number

Pilot
 Instruments onboard
 Number of data channels
 Number of digitization levels per data channel
 Spectral bands assigned to each data channel
 Instrument gain for each channel
 Data recording start time
 Data recording stop time
 Number of scan lines recorded
 MAS Internal/GOES clock flag
 INS data availability flag
 INS Internal/GOES clock flag
 Instrument operator (gain setup, calibration)
 Comment (e.g. data quality, channel dropouts)
 Number of Level-0 input tapes
 Number of Level-1 output tapes
 Location of Level-0 input tapes (with contact)
 Location of Level-1 output tapes (with contact)

■ *LINE* Table

Granule ID
 Mission Number
 Flight Number
 Line Number
 Start Time
 End Time
 Duration
 Nominal heading
 Nominal altitude
 Nominal spatial resolution
 Number of scan lines
 Start scan line number
 End scan line number
 Nadir start latitude
 Nadir start longitude
 Nadir end latitude
 Nadir end longitude
 Nadir start solar zenith angle
 Nadir start solar azimuth angle
 Nadir end solar zenith angle
 Nadir end solar azimuth angle
 Day/night flag

Land/water flag
Cloud/clear flag
Snow/clear flag
Nominal blackbody #1 reference temperature
Nominal blackbody #2 reference temperature
Browse/quick-look file name
Level-0 tape ID
Level-1 tape ID
Location keyword
Comments (data quality, coverage features)

■ *WHO* Table

ID Number
Last name
First name
Institution
Title
Address
Phone
Fax
Email
Comments

■ *REFERENCE* Table

Mission Number
Reference ID
Author(s)
Title
Citation
Abstract (free text)
Keyword 1
Keyword 2
Keyword 3

Retrieval and/or visualization of browse/quick-look products

It is often useful for a prospective data user to be able to view a subsampled portion of a data set before retrieving the entire data set. For example, a particular user might be interested in flight lines which have snow cover, and no clouds. There are several ways in which this can be accomplished.

The first method would involve the creation of subsampled images for each flight line as part of the Level-1 processing. These images could be subsampled both spectrally (one visible and one infrared channel) and spatially (every fourth pixel on every fourth scan line), and stored in some portable image format (e.g. Graphics Interchange Format, GIF). These images would be created for every flight line, and have IDs corresponding to the flight line IDs. They would be available for download (by FTP) to the user's own local machine where they could be viewed using freely available imaging software (which would be provided, or at least instructions would be given on where to obtain such software). This method is the simplest and most effective for users who access the MAS catalogue system remotely over systems such as Internet. Since each user will have a local imaging program which reads the standard image format, no imaging interface need be provided by the catalogue system itself.

Another method would be to allow the user to create subsampled imagery in real-time from existing data sets, and to display these using some graphics standard tool provided by the catalogue system. This is certainly a more complicated system to design, as it requires

- a complete library of data sets online which the user can access,
- sufficient computer resources for the user to create browse images as desired,
- a standard graphics interface which the user can handle (e.g. X Windows).

In the near term, it appears that the first method would suffice. The main advantage is that it would require the minimum of effort for both users, and for the setting up of the catalogue system. User effort is an important consideration, since users will not want to spend an inordinate amount of time or money setting up their own system just so they can view browse products. The first method would also be considerably easier to implement at present, as it would require only a modest effort to create subsampled images routinely as part of the MAS Level-1 processing, and make these images available by FTP.

Recommendations

The first version of the MAS catalogue system should contain the following features.

The system would be located on a system that is accessible via Internet. Users would be able to log on from Internet nodes simply by giving the correct username. A first time user would be asked to enter details such as name, address, affiliation etc. and would be asked to keep this information updated.

The user interface would be character based, and compatible with several standard terminal protocols (e.g. DEC VT-100, ANSI). A simple list of menu options would be presented, with courses of action as follows:

- (1) Review MAS data guide (Instrument description and mission schedule)
- (2) Query MAS data inventory (using DBMS)
- (3) Select and download (by FTP) MAS browse/quick-look imagery
- (4) Place request for MAS data copies

Depending on the level of complexity of the catalogue system, each of these levels would be linked, so that for example the results of an inventory search could be used to select and download browse/quick-look imagery.

Investigation of strategies for storing and distributing MAS data

Liam Gumley, RDC

20 February 1992

A first version of the MAS Level-1 processing system is now operational. Data from a field campaign is being processed and distributed to users. However for future data handling, it is necessary to implement some strategy for storing and distributing the MAS data in an efficient manner. To assess the requirements for a data storage and distribution system, the current and anticipated future MAS data volumes are examined.

The MAS has at present flown one complete science mission (FIRE). The next mission will be ASTEX (June 1992), followed by a biomass burning mission in the Amazon in September 1992. No dates have yet been confirmed for further missions.

Present MAS Level-0 data volume status

The MAS Level-0 data is currently received from Ames Research Center on 9 track 6250 bpi magnetic tapes. The data contained on these tapes is summarized as follows:

Tape type:	9 track 6250 Bpi 2400 ft
MAS scanlines/tape:	18800
Bytes/scanline:	10320
Bytes/tape:	18800 scanlines/tape * 10320 bytes/scanline = 194.016 MB
MAS scan rate:	6.25 scans/second
MAS data rate:	6.25 scans/second * 10320 bytes/scanline = 64500 bytes/second = 232.2 MB/hour
MAS Flight time:	Up to 8 hours (maximum)
<u>MAS data/flight:</u>	8 hours * 232.2 MB/hour = <u>1857.6 MB</u>
<u>MAS tapes/flight:</u>	1857.6 MB / 194.016 MB/tape = 9.574468 tapes ≈ <u>10 tapes</u>
INS sampling rate:	1 sample every 5 seconds
Bytes/sample:	150
INS data rate:	150 bytes every 5 seconds = 30 bytes/second = 108000 bytes/hour = 0.108 MB/hour
<u>INS data/flight:</u>	8 hours * 0.108 MB/hour = <u>0.864 MB</u>
<u>INS tapes/flight:</u>	≈ <u>1 tape</u>
Flights/mission	= 1 to 20 typically
<u>MAS data/mission</u>	= (1 to 20) * 1857.6 MB = 1.8576 GB to 37.1520 GB
<u>MAS tapes/mission</u>	≈ (1 to 20) * 10 tapes = 10 to 200 tapes
<u>INS data/mission</u>	= (1 to 20) * 0.864 MB = 0.864 MB to 17.280 MB
<u>INS tapes/mission</u>	≈ (1 to 20) * 1 tape = 1 to 20 tapes

Future MAS Level-0 data volume status

By June 1992 it is planned to have an Exabyte tape data system on board the ER-2. It is likely that this will mean MAS Level-0 data will be distributed on Exabyte tape. An Exabyte 8200 tape has a nominal storage capacity of 2 GB. Thus it should be possible to hold a complete set of MAS Level-0 data from one flight on one Exabyte tape (using the present 12 channel data system).

Sometime in the near future (exact date not yet known) the MAS data system is expected to be upgraded to 50 channels @ 12 bits/channel. This would increase the output data rate by a factor of approximately 6.25 to 1451.25 MB/hour = 1.45125 GB/hour. Likewise the total volume of data from a flight would increase to a maximum of 11.61 GB. The amount of data from a mission would then range from 11.61 GB to 232.2 GB.

No significant increase is expected in the INS data volume in the near future.

Present MAS Level-1 data volume status

MAS Level-1 processing generates calibrated, geolocated radiances from the Level-0 MAS and INS data streams. The radiances are stored as scaled 16 bit integers, and comprise the largest component of the output data stream. Currently the Level-1 MAS sets are about 2 times as large as the corresponding Level-0 MAS data sets. However only straight flight tracks (lines) are processed to Level-1. Data taken during turns or ascent/descent is not processed. The fraction of a flight which is made up of straight lines is usually between 50% and 75% of the total flight time. If 75% is a typical maximum value, then the total size increase from Level-0 to Level-1 is around $2 * 0.75 = 1.5$.

MAS data rate: $\approx 232.2 \text{ MB/hour} * 1.5$
 $= 348.3 \text{ MB/hour (at Level-1)}$

MAS data/flight: $8 \text{ hours} * 348.3 \text{ MB/hour (maximum)}$
 $= 2.7864 \text{ GB (at Level-1)}$

MAS tapes/flight: $\approx 2 \text{ tapes (Exabyte 8200)}$

MAS flights/mission: $= 1 \text{ to } 20 \text{ typically}$

MAS data/mission: $\approx (1 \text{ to } 20) * 2.7864 \text{ GB}$
 $= 2.7864 \text{ to } 55.7280 \text{ GB (at Level-1)}$

MAS data/mission: $\approx 1 - 28 \text{ tapes (Exabyte 8200)}$

It can be seen that Exabyte 8200 tapes, with a storage capacity of around 2 GB are a natural choice for storing the Level-1 data sets. Nine track 6250 Bpi 2400 ft magnetic tapes can hold a maximum of 180 MB, and since many of the individual flight line files at Level-1 are larger than this, these tapes are not suitable for storage of large volumes of MAS Level-1 data.

Future MAS Level-1 data volume status

The MAS Level-1 data sets are stored in the Network Common Data Format (netCDF). This format is capable of handling 8, 16 and 32 bit integers, as well as 32 bit floating point numbers. In determining the data types used for storing the various items in the MAS Level-1 data set, a balance was struck between storage precision, and efficient space usage.

In order to minimize the size of the output data sets, it was therefore decided to store all the radiance data as scaled 16 bit integers, rather than 32 bit floating point numbers. Initial tests showed that this would retain sufficient precision in the radiance data, as well as keeping the data set size down. It was also decided to store geolocation data for every tenth pixel on every scan line, in order to save space.

As the usage of the MAS data increases, it may be deemed necessary to store the radiances in 32 bit floating point format, or to store geolocation data for every pixel on every line. At present, for each scanline of output data in the Level-1 dataset, around 86% of the data is the radiance information, and 9% is the geolocation information. It can be seen that increasing the number of bits stored for radiance information from 16 to 32 would almost double the size of the output data set. Similarly, storing geolocation information for every pixel rather than every tenth would also almost double the size of the output data set. If both changes were made, the dataset size would increase by a factor of $(2 * 86\% + 10 * 9\%) = 262\%$. Thus the total volume of data from a flight would increase to a maximum of $(2.7864 \text{ MB} * 2.62) = 7.3004 \text{ GB}$. The amount of data from a mission would then range from 7.3004 GB to 146.0073 GB.

The use of a 50 channel data system would also impact the Level-1 data set size. The use of 12 data bits per channel would not affect the output data set size since space is already allocated for at least 16 bits per channel of radiance data. An increase in the total number of channels from 12 to 50 would increase the Level-1 data set size by a factor of about $(50 / 12) = 4.167$. Therefore the maximum amount of Level-1 data from a flight would be $(4.167 * 2.7864 \text{ GB}) = 11.6109 \text{ GB}$. The total amount of Level-1 data from a mission would then range from 11.6109 GB to 232.2186 GB.

Level-0 data storage and distribution requirements

Currently all MAS and INS Level-0 data is delivered directly to RDC on 9 track tapes. The total number of these tapes received so far is 58, mostly from the FIRE mission in November/December 1991. It is reasonable to assume that approximately this number of tapes will be generated for each MAS mission (i.e. between 10 and 200). The tapes are currently logged and stored on shelves at RDC, and are taken to GSFC in lots of 3 to 5 to be read and processed. Clearly a storage facility at GSFC would be preferable for storing the Level-0 tapes, integrated with the MAS processing facility if possible. At a minimum this would require several sets of tape racks in a reasonably secure area. Some mechanism would also have to be out in place to record and track the whereabouts of the tapes if they are moved from this area. Since there are not likely to be any users of the Level-0 data apart from the Level-1 processors, distribution should not be a major concern.

Level-1 data storage and distribution requirements

The Level-1 data will be required by several different users. Up until the present, the MAS users group has been limited to MODIS team members on site at GSFC. MAS data has been available primarily by anonymous FTP.

In future, it is likely that more members of the MODIS Science Team will require access to MAS Level-1 data, as well as some other EOS investigators and external researchers. Given that a catalogue system will exist where users can select portions on a MAS flight, then some means of copying and distributing this data needs to be put in place.

For example, if 5 MODIS team members and 3 external researchers require access to data from one MAS flight, each requiring a different set of 5 flight lines from a total of 20, then each user will need their own specially selected copy of the data set. A simpler possibility would be to give any user who requests data from a flight ALL of the Level-1 data produced for that flight. A scenario is as follows.

Number of MAS users:	= 8
Requests/user/flight:	= 1
Requests/user/mission:	≈ 1 to 20 (assuming 1 to 20 flights per mission)
<u>Possible requests/mission:</u>	<u>≈ 8 to 160 requests</u>
MAS Level-1 data/mission:	= 2.7864 to 55.7280 GB
<u>Distributed data/mission:</u>	<u>= (2.7864 to 55.7280 GB) * 8 users @ 100% Level-1 request</u> <u>= 22.2912 to 445.824 GB</u> <u>= 11.1456 to 222.912 GB @ 50% Level-1 request</u>
Data/user/mission:	= 2.7864 to 55.7280 GB @ 100% Level-1 request ≈ 2 to 28 Exabyte 8200 tapes ≈ 1 to 14 Exabyte 8200 tapes @ 50% Level-1 request
<u>Distributed tapes/mission:</u>	<u>≈ (2 to 28 tapes) * 8 users @ 100% Level-1 request</u> <u>= 16 to 224 Exabyte 8200 tapes</u> <u>= 8 to 112 Exabyte 8200 tapes @ 50 % Level-1 request</u>
Tape copy time/tape:	≈ (2 GB / 15 MB/minute) (optimal Exabyte 8200 tape write) = 133.33 minutes = 2.22 hours per Exabyte 8200 tape
Distribution time/tape:	≈ (2.22 hours copying + 1.0 hours setup/mailling) (optimal) = 3.22 hours per Exabyte 8200 tape
<u>Distribution time/mission:</u>	<u>≈ (16 to 224 tapes) * 3.22 hours/tape</u> <u>= 51.52 to 721.28 hours @ 100% Level-1 request</u> <u>= 25.76 to 360.64 hours @ 50% Level-1 request</u>

Level-1 data distribution by FTP

Currently the Level-1 data is generated on the LTPIRIS2 system, and about one flight of data may be held at any one time (2 1.2 gigabyte disks are available). Users may retrieve portions of the flight by anonymous FTP. The optimum transfer rate onsite at GSFC is 1.25 MB/second, and in most situations the transfer rate will be less than 80% of this figure. Users external to GSFC will typically experience transfer rates of less than 0.5 MB/second. In any case, using FTP for data retrieval implies that

- (1) data from one flight be copied to the appropriate disk area,
- (2) users be notified that the data is available,
- (3) users download the data to their local systems,
- (4) the data removed and step (1) repeated.

It should also be noted that having more than one user downloading data at one time will slow down the transfers significantly.

FTP Transfer time:

$$\begin{aligned} &\approx (22.2912 \text{ to } 445.824 \text{ GB}) / (0.8 * 1.25 \text{ MB/second}) \\ &= (22.2912 \text{ to } 445.824 \text{ GB}) / (0.001 \text{ GB/second}) \\ &= 22291.2 \text{ to } 445824.0 \text{ seconds} \\ &= \underline{6.192 \text{ to } 123.840 \text{ hours (@ 100\% Level-1 request)}} \end{aligned}$$

This transfer time assumes a user onsite at GSFC who does NOT pause during transfer to archive the data at his local site i.e. it assumes the user has sufficient space to store ALL the downloaded data. Obviously, few users will be in this situation. The corresponding times for users offsite will be considerably higher. Thus it appears that FTP is not an appropriate distribution mechanism for handling routine distribution of MAS data, apart from small special cases for users onsite at GSFC.

Level-1 data distribution on magnetic tape

The only reasonable option for Level-1 data distribution is high-density magnetic tape. 9 track 6250 Bpi tapes (180 MB maximum capacity) do not meet the high-volume requirements of MAS data.

Exabyte 8200 tapes appear to be the most useful storage/distribution media. Ideally, a system for storing and distributing the MAS Level-1 data would therefore have the following features.

- (1) A fast CPU for MAS processing.
- (2) \approx 5 GB of disk storage (half for processing, half for distribution).
- (3) A minimum of two Exabyte 8200 tape drives for tape copying. A 'hopper' system which could copy multiple tapes would be more useful.
- (4) Links to the MAS data catalogue to determine data request requirements.
- (5) Local archive facilities to maintain copies of all MAS Level-1 data (to respond to future requests without re-processing).
- (6) Sufficient operator coverage to allow data requests and catalogue/archive functions to be handled efficiently.
- (7) Internet links to correspond with users.

It is suggested that the MAS archive system be collocated with the MAS catalog system to maintain the flow of information between these two functions. Separating the two would increase the difficulty of maintaining an accurate catalogue.

**Cloud Algorithm Porting
MODIS SDST
Thomas E. Goff
20 February, 1992**

tgoff on GSFC mail,
teg@ltpiris2.gsfc.nasa.gov,
or (301) 982-3704

Lessons Learned from the CLDOPT program port.

History: This program is currently executing on the GSFC IBM mainframe computers and is used to perform corrections to the MCR instrument data. It was written in 1988 and 1989 in IBM FORTRAN 77 by T. Nakajima who is no longer at GSFC. It consists of 1814 lines of code and utilizes nine input datasets and one output dataset as tabulated below:

<u>name</u>	<u>size (bytes)</u>	<u>lrecl</u>	<u># of recs</u>	<u>type</u>
dlib1	891	80	11	EBCDIC
cmprst	561792	2112	266	mixed
ofhom1m	26892	80	332	EBCDIC
ofhom2m	26892	80	332	EBCDIC
ofhom3m	26892	80	332	EBCDIC
oasymp1	72819	80	899	EBCDIC
orhom1	4548960	4320	1053	REAL
orhom2	4548960	4320	1053	REAL
orhom3	4548960	4320	1053	REAL
orsemi	4548960	4320	1053	REAL
ocptrm	209088	2112	99	mixed

Code Porting: The FORTRAN code was FTP'd from the IBM machine to the UNIX machine in ASCII mode to perform an automatic conversion of the IBM EBCDIC characters to ASCII. The code then compiled with no errors but contained several warnings about uninitialized data arrays and variables, and unused symbols.

Data Porting: The EBCDIC files needed to be converted to ASCII - the UNIX FTP facility performs this in it's ASCII mode of file transfer. The REAL type files needed converting from IBM floating point form to UNIX floating point form. This was accomplished (knowing the magnitude of the numbers and accepting some loss of precision) by multiplying each number by 100000.0 and storing in place as integer numbers. The file is now in 2's complement form that can be FTP'd in binary mode to UNIX, then converted back into native floating point form. The IBM machine and the UNIX machine do not require byte swapping. The same technique was applied to the mixed type files, but required more work to insure that the equivalence specifications were correct. One of these files has a double hidden equivalence that resulted in additional file snooping to find the problem. Two FORTRAN programs were written on the IBM machine and two programs were written on the UNIX machine to accomplish this data port.

Utilities Written: In order to expedite the above tasks, several utility and library functions were written. These included the fdump - file dumping utility to determine the data values of the EBCDIC, ASCII, integer, and floating point values within each file. Every data value could not be verified due to the large size of these files. The original program used the IBM FTIO package to read and write the mixed (binary) type file to and from the datasets on the IBM disc. In order to not touch the original program and treat this problem as a computer porting effort instead of an

algorithm writing effort, a library (ufio) was written to emulate these unformatted binary I/O functions in the UNIX system. UNIX generally does not have a concept of a file record while the IBM machine does. Therefore, this library emulates the fixed record size reads and writes of the IBM machine but requires that the user preload the library utilities with the correct record sizes as shown by the IBM map command.

Program Execution: The program with the above libraries begins execution on the UNIX machine but eventually aborts with either a bus or segment error as shown:

```
cldopt <fort.5
* MCR-DATA ANALYSIS *
Unit, FileName, RecSize, Status: 10, fort.10, 2112, rb
  1 : NFL AG IDP/ PARAMETERS
    0.060 0.050 0.045
      10      87      7      13      18      28      55      559 51453
1
      6      6      161      0      04266700      0      0 1525*****
Unit, FileName, RecSize, Status: 30, fort.30, 4320, rb
Unit, FileName, RecSize, Status: 31, fort.31, 4320, rb
Unit, FileName, RecSize, Status: 32, fort.32, 4320, rb
Unit, FileName, RecSize, Status: 33, fort.33, 4320, rb
Segmentation fault (core dumped)
%
```

This output agrees with the IBM output up to the 15th value in the above integer array. The remaining values on the IBM are 0's but are obviously garbage in the UNIX machine. This is indicative of non-initialized arrays that the IBM machine sets to 0, but the UNIX machine does not. This is probably the cause of the abend and subsequent core dump. The above output includes messages from the utility library verifying the opening of the binary data files with the correct record lengths.

Man Power Estimates: The porting effort up to this point has taken approximately one (1) man-month. This includes the writing of utilities, but not the full configuration management and peer reviews of those utilities. The majority (75%) of the time was spent in understanding and dealing with the infrastructure of the various computers involved, especially the IBM machine. A list of obstacles would include:

- Access to the main frame - Key board mapping for IBM, direct or Telnet access to the IBM. Direct dial allows only the IBM line editor, QED, to be used. Telnet allows the screen editor if you can get telnet to talk intelligently to EBCDIC machines (the VAX telnet works, UNIX telnet does not, TN3270 core dumps), and also perform the correct keyboard mappings and character translations for the IBM PK and PA keys. These are necessary to exit any task on the IBM from telnet!
- Time delays in fetching datasets, editing jcl, and executing jobs on the IBM machine. This machine was in the process of being updated during this effort which resulted in trying to "hit a moving target" with jcl and operating system commands.
- Investigating the IBM FTIO subroutine package and the determination of the IBM dataset member record lengths.
- Creating dataset members on the IBM machine with the correct track allocation in addition

to the dataset specifiers.

- Dumping dataset contents on the IBM machine to determine the magnitude of the data values. (The radiance values for the MCR instrument were normalized.)

The remainder of the time (25%) constituted the writing of the programs to transform the datasets on both the IBM and UNIX machines, writing the unformatted file I/O library on the UNIX machine, and verifying the dataset contents on both machines.

Futures: Further work to complete this port would include the debugging of the science code. Before additional work is performed on this code, it is recommended that this program be completely revised to conform to good coding standards with adequate documentation and references added. Using 1.5 to 3 lines of code (LOC) per hour results in a 1.3 to 2.6 man-year estimate to bring this program up to good programming standards.

Recommendations for Science Code: Several items that would facilitate the porting of science code in the MODIS era would include:

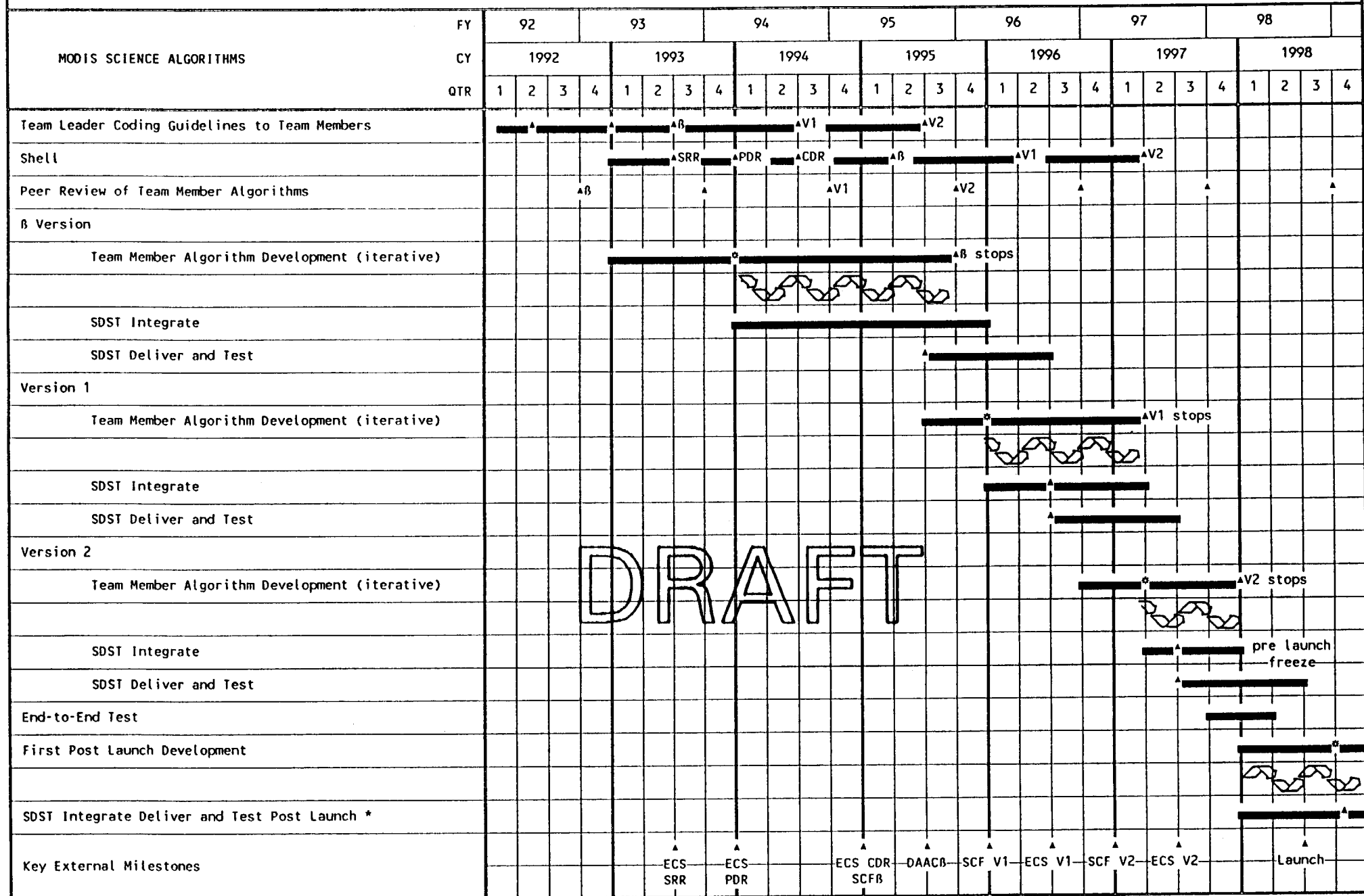
- * A full compilation of the code with all warnings enabled and exercised before the port is attempted. This includes the initialization and full typing of all variables.
- * All code should be reviewed for acceptance to project code standards in addition to a peer review of the computer algorithms involved.
- * All datasets and programs that create the datasets should be generated on the same (or fully compatible) machine where the algorithm is to be executed. This would eliminate the need for floating point data conversion, byte swapping, or character conversion.
- * UNIX does not fully support the concept of a logical record, just a stream of Bytes. Code that takes advantage of an abbreviated record read (requesting n bytes from an m byte record where $m \gg n$) should not be allowed. This is especially true for record reads into sparse arrays (arrays dimensioned larger than the data set record). Logical record sizes should be totally contained within the data set: i.e. possibly using the first word of the data set as the record length in a run length scheme.
- * Utilize or invent a method for the dynamic allocation of memory. This allows arrays of variable size to be allocated as need with changes to the source code. A single program could be written that could handle remotely sensed, two dimensional (or greater) arrays in which the dimension sizes can be specified as user inputs rather than hard coded into the program.
- * Datasets that contain mixed data types (I*2, I*4, Character, R*8, etc) should preferable not be mixed in the same file. An alternate to this, is to write a library that saves datasets in a run length scheme. This would allow datasets to be passed among machines that do not support user specified fixed length record lengths. Mixed modes and differing floating point bit lengths can be included. This facility is possible with the NetCDF data interchange format, provided these libraries can be implemented and supported on all machines including the IBM main frames. Note that NetCDF does not currently support mixed data types within a variable array.
- * A port of and update to the cldopt program to the Cray computer has been started. The

Cray native floating point default is 64 bits which is incompatible to 32 bit defaults on other machines. Datasets with mixed mode types would be need to be modified to accomplish this port when 32 bit floating point has been assumed.

- * All file specifications should be performed using code in the source program, not the jcl. This makes the file specifiers more machine independent. Use the FORTRAN and C "open" statements. If this is impossible to accomplish due to differing file directory specifiers (i.e. dir.sub.file.ext in VAX vs. dir/sub/file.ext in UNIX), than a library shell to accomplish this task needs to be provided. This philosophy also applies to file reads and writes.

MODIS Science Data Support Team (SDST) Schedule

ge 1



Notes: This schedule assumes continuous interaction between ECS and SDST and that formal ECS reviews (SRR, PDR, CDR, etc.) are NOT the only avenue for information exchange. Team members deliver code, algorithmic tables and data sets, test data, test results, and documentation.

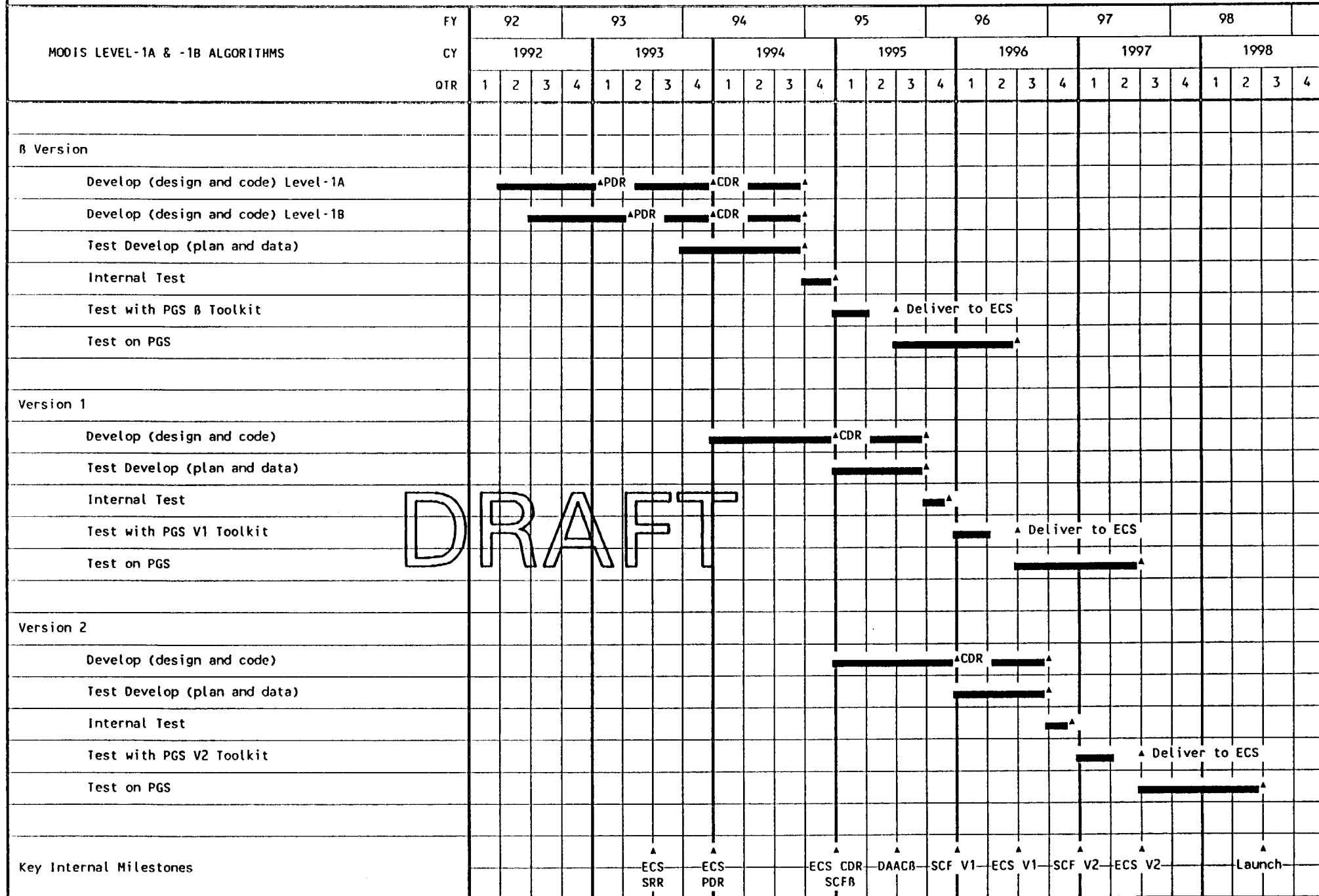
* Turnaround at SDST and ECS 1 - 2 weeks.

* First Delivery

DRAFT

MODIS\PLANS\GHANT.08B

February 12, 1992



Note: This schedule assumes continuous interaction between ECS and SDST and that formal ECS reviews (SRR, PDR, CDR, etc.) are NOT the only avenue for information exchange.
 MODIS\PLANS\GHANT.088

February 12, 1992

MODIS Science Data Support Team (SDST) Schedule

Page 4

ACTIVITY	FY	92				93				94				95				96				97				98			
	CY	1992				1993				1994				1995				1996				1997				1998			
	QTR	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
TLCF																													
Requirements Definition																													
Develop Evolution Strategy																													
Team Leader Computing Facility Plan																													
Design																													
Procurement																													
Utilities																													
Utilities for Use with MAS or MODIS																													
MetaData																													
Browse																													
Location/Matching to In Situ																													
Other, e.g. Data Conversion, Resampling/Subsampling																													
Input Data																													
From Other Instruments																													
From MODIS Tests																													
From Simulations																													
For Development																													
Valid for Algorithm Examination																													
Defective for Algorithm Testing																													
For Validation/QC																													
Auxiliary for Processing																													

TBS

